Project: Streaming Synchronized Multimedia (SSMM) Project

Project Title and Leaders

Wo Chang (wchang@nist.gov)

Chronological Summary and Date of Last Update

1. March 12, 1998 - Project proposal under development

Executive Summary

Streaming multimedia over the Intranet/Extranet/Internet becomes an ever-critical requirement for many corporations, especially in the areas of electronic commerce, corporate teleconferencing, long distance learning, and self-paced training. This technology offers live multicast play out and an instant replay (for archived clips) to play the audio, video, periodic-update background still-image and text-caption rather than the traditional approach, which requires downloading the entire clip before playing it, which is very time consuming and ineffective. Within streaming technology, it is desired to dedicate each media type to its own media stream for the purpose of network adaptability but still provide single cohesive presentation to the end user by synchronize all the incoming streams.

This project proposal outlines the architecture and technical approach for creating streaming synchronized multimedia application and its testbed. It uses Internet Engineering Task Force (IETF) standard protocol technologies such as HTTP, FTP, and RTP for media streaming; adopts RTSP for control and manage media streams; and applies multi-layer-streaming scheme for network bandwidth QoS;. For content presentation, this project will use World Wide Web Consortium (W3C) industry defined SMIL (Synchronization Multimedia Integration Language) markup language to create time-based and temporal related multimedia content. The primary objective of this basic research and development of streaming synchronized multimedia will involve NIST to create reference implementation of IETF standard real-time transport protocols and W3C content presentation technology for the area of Intranet/Extranet/Internet multimedia and digital video technology to the public.

Problem Description

As the web continues to grow at its rapid rate, corporations try to apply web-based multimedia technology to streamline their daily businesses, especially in the areas of online telemarketing, virtual mall shopping, entertainment sectors such as video-on-demand and game-on-demand, and telecommuting via desktop remote conferencing. Traditionally, the above applications usually use the standard codec like MPEG, which combines the audio and video into a single media stream to transmit the data to the client side. This approach has the following major bottlenecks:

- *Flexibility* – since the audio is bound to the video content, there is no way to separate the media stream so that different language audio streams can be transmitted separately while multicasting a single video stream.
- *Efficiency* – since both audio and video streams are encoded into a single stream, there is no way to adjust the video resolution dynamically so the quality audio may continue to be transmitted.
- *Scalability* – since both video and audio streams are encoded into a single stream, sending multiple same-video (with different-audio) streams would easily congest the network.

With the bottlenecks mentioned above, the logical solution is to separate all media types into its own media stream and let the client application synchronize them into a single cohesive presentation. In order to achieve this goal, the following Internet-standard technologies shall be investigated and implemented:

- Apply IETF standard protocol components such as HTTP, FTP, and RTP for streaming the media, adopt RTSP for control and management of the streams, and apply multi-layer-streaming scheme for network bandwidth QoS.
- Apply SMIL (Synchronized Multimedia Integration Language) specification of Synchronized Multimedia working group of W3C for content composition.
- Use standard ISO and ITU media types such as MPEG-2, H.261, MPEG-4, and possible MHEG-5 as the RTP-based streaming payload formats.
- Use industrial (Sun, SGI, and Intel) defined Java Media Framework as a framework to develop Java applet for cross-platform deployment.

NIST Objectives and Statement of Work

<u>NIST Objectives</u>

The main objective of this project is to prototype a platform independent streaming synchronized multimedia application and create the needed testbed, so that a public reference implementation of integrated W3C SMIL and IETF RTP/RTSP based multimedia player can be prototyped. This prototype will allow researchers to (a) have an environment to study various synchronization techniques and (b) have a tool to measure various media type streams performance characteristics. It is important to have an environment and tools to test out the efficiency of specific leading edge technology in the areas of SMIL, RTSP, and RTP.

<u>Statement of Work</u>

For FY-98, SSMM will mainly concentrates on SMIL, synchronization, RTSP, and RTP. With the SSMM architecture block diagram shows below (Figure 1), it is important to develop an incremental scheme to implement the JMF-based SSMM applet. The first step is to get the basic model working, which is to implement SMIL parser and event synchronizer. So that the SSMM applet can retrieve HTTP-/FTP-based content presentation page and be able to understand and play out the temporal and spatial aspects of that page accordingly. The second step is to expand the transport protocols from HTTP/FTP with RTP, and use MPEG-2/H.261 as RTP-based payload formats along with RTSP stream control. If time permits within this fiscal year, the multi-layered-streams scheme and more RTP-based payload formats (MPEG-4 and MHEG-5) should then be implemented.

Figure 1: Streaming Synchronized Multimedia Architecture

<table>
<tr><td colspan="4">SMIL[12]</td></tr>
<tr><td colspan="4">Application Control (RTSP[1], etc)</td></tr>
<tr><td colspan="4">payloadware[2]</td></tr>
<tr><td>MPEG-2[3]</td><td>H.261[4]</td><td>MPEG-4[5]</td><td>MHEG-5[6]</td></tr>
<tr><td colspan="4">Multi-layerd-streams[7]</td></tr>
<tr><td colspan="4">HTTP/FTP/RTP[8]</td></tr>
<tr><td colspan="4">QoS mapping
(RSVP[9], ATM signaling[10], etc)</td></tr>
<tr><td colspan="4">middleware[11]</td></tr>
<tr><td>ATM</td><td>IP/6</td><td>IP/4</td><td>DAVIC</td></tr>
</table>

Figure 2 shows the SSMM testbed infrastructure, which comprises of a Unix multimedia platform running Sun Solaris, a multimedia Window NT server, a Window 95 workstation, and a Window 95 running either Netscape/Internet-Explorer browser. This testbed provides a development environment and a facility to conduct various performance measurement testing. As for testing, it allows us to send a single integrated audio and video stream like MPEG-2 from one station as well as sending one video stream from station A, while station B sends the audio stream, and station C only sends the periodic-update background still image and a text-caption stream. The client browser simply receives all the streams and synchronizes them. In addition, since SSMM applet is multicast enable, it can act as an integrated (audio and video) player to play Internet MBone live multicast audio and video, the same way as with VIC and VAT except that this is all done within one single application, the SSMM.

Figure 2: Streaming Synchronized Multimedia Testbed

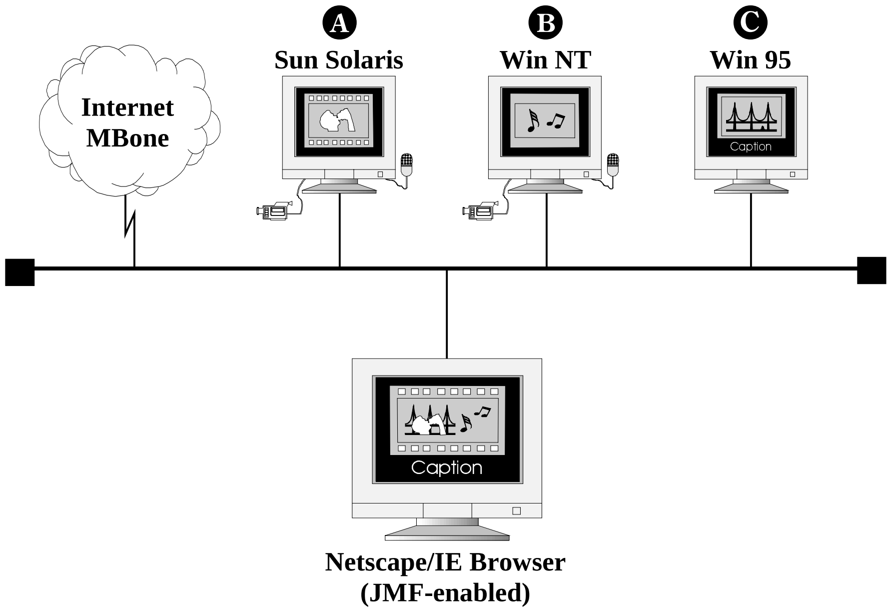
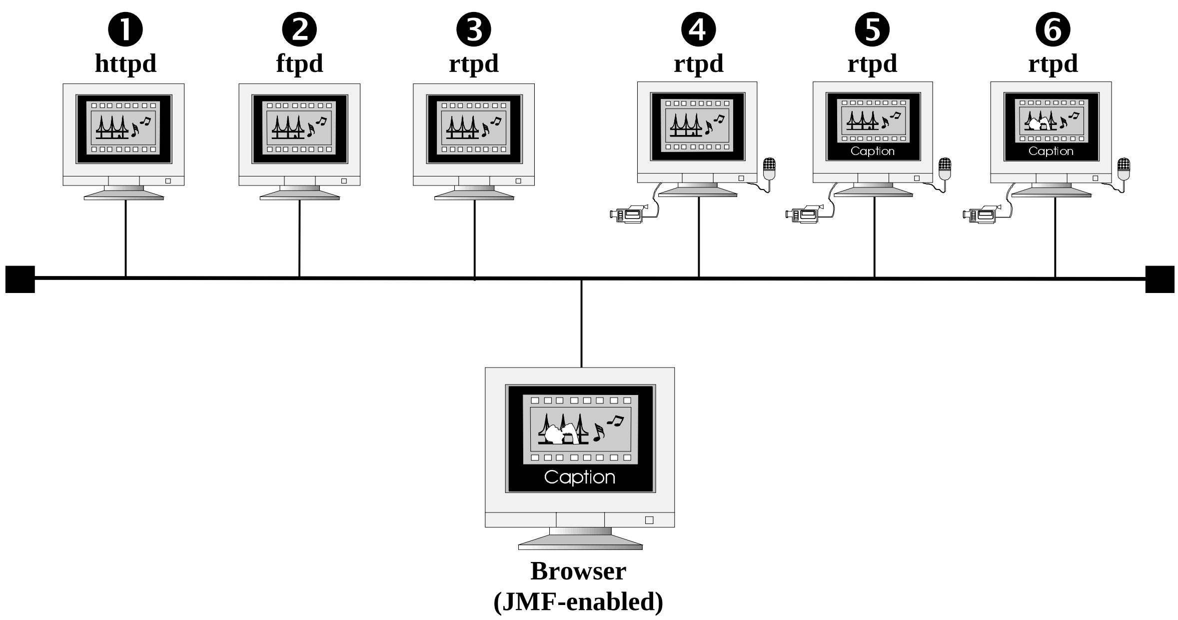


Figure 3 shows a possible combination of performance measurement scenarios. Each scenario uses the SMIL content markup language to layout the time-based and temporal-space relationship for its presentation. The followings are some interesting performance measurement scenarios, especially measuring from within a congested network environment:

Figure 3: SSMM Performance Measurement Scenarios



❶ MPEG-2 Audio/Video over HTTP Protocol

Sending single integrated MPEG-2 audio/video stream from a httpd web server with its http protocol. Measuring the effective of the transmitting time as well as the packet lost of the stream. Use this measurement as the base reference when comparing to the separate media type streams.

❷ MPEG-2 Audio/Video over FTP Protocol

Sending single integrated MPEG-2 audio/video stream via the ftp server. Measuring the effective of the transmitting time as well as the packet lost of the stream. Use this measurement as the base reference when comparing the separate media type streams.

❸ MPEG-2 Audio/Video over RTP Protocol

Sending single integrated MPEG-2 audio/video stream via rtp protocol. Measuring the effective of the transmitting time as well as the packet lost of the stream. Use this measurement as the base reference when comparing the separate media type streams.

❹ Separate Audio/Video Streams over RTP Protocol

Sending dedicated separate audio/video streams via rtp protocol. Measuring the effective of the transmitting time as well as the packet lost of the stream. Compare the result with the single integrated audio/video stream from http/ftp/rtp protocols.

❺ Separate Audio/Video/TextCaption Streams over RTP Protocol

Sending dedicated separate audio/video/text-caption streams via rtp protocol. Measuring the effective of the transmitting time as well as the packet lost of the streams. With the extra overhead from the text-caption, compare the result with the single integrated audio/video stream from http/ftp/rtp protocols.

❻ Separate MPEG-4-Object/Audio/Image/TextCaption Streams over RTP Protocol

Sending single moving MPEG-4 object along with periodic-update static background image and a pair of separate audio/text-caption streams via rtp protocol. Measuring the effective of the transmitting time as well as the packet lost of the streams. With the reduced video stream but extra overhead from the periodic-update static image and text-caption stream, compare the result with the single integrated audio/video stream from http/ftp/rtp protocols.

Our tentative schedule and deadlines are as follows:

December 30, 1997: Preliminary research and evaluation
February 28, 1998: Detailed design and refinement
March 30, 1998: Established SSMM testbed
April 30, 1998: Completed SMIL parser and its event synchronizer
May 30, 1998: Completed MPEG-2 stream over HTTP/FTP/RTP
August 15, 1998: Completed RTSP stream control and management
August 30, 1998: Completed initial performance measurement for scenario 1 to 5
September 30, 1998: Publications of results & release prototype to public

Rationale for NIST Participation

NIST's participation in this task is based on our ITL mission, which is to actively participate relevant leading edge information technology at its early infancy stage and to assess if the new technology will benefit the industry and corporation. SSMM project fits this criteria perfectly.

In addition, there are other reasons for pursuing this task:

- Establish technical expertise and measurement tools for NIST to make future contributions to the design and development in the area of multimedia and digital video,
- Establish NIST reputation through the working relationship with other standard bodies and leading multimedia software and digital video vendors.

Required NIST Resources

This task requires 2.5 full-time staff members to complete the SSM initial prototype. It will be necessary for at least one, if not both, members of the effort to become active participant of the relevant IETF, W3C, MPEG-4, and DAVIC working groups. This will require attendance at 4-6 working group meeting/year (usually couple meetings per year are head outside the US).

The SSMM prototype development will require one dedicated multimedia Sun workstation as the audio/video stream server (with minimum of 64 Mbyte RAM memory and 8 Gbyte hard disk), along with three high performance multimedia PCs (at least Pentium II MMX with SVGA monitor, 64 Mbyte RAM, 8 Gbyte hard disk, video camera, and audio speaker) for one Window NT server for sending separate audio/video streams, one Window 95 workstation for sending periodic-update background still image and text-caption steam, and one Window 95 workstation for browser client.

External Resources and Collaborations

With fast emerging standard technologies such as multimedia content presentation (SMIL), content rich payload formats (MPEG-4 and MHEG-5), cross platform tools (JMF), and network transport protocols (RTP and RTSP), it is important for NIST to evaluate and to adopt. It is even more critical for NIST to collaborate with these standard organizations to keep up with the technology direction as well as become an active participant in these communities.

Criteria for Task Evaluation / Risk Factors

Streaming synchronized multimedia is at its very early infancy stage, particularly when working with multiple technology areas: Internet standard RTP and RTSP protocols; MPEG-4 and MHEG-5 multimedia payload formats, JMF cross-platform Java language, and SMIL content composition markup language. There are many risk factors involved to make the SSMM project to be a successful one. However, since these standard bodies such as IETF, W3C, and overwhelming industrial support for JMF (Sun Microsystems, SGI, and Intel), the chance for NIST to success in this project should be pretty high. JMF betas are already available from Sun, SGI, and Intel, and are running under operation systems included Windows 95, Windows NT, Sun Solaris, and SGI Irix platforms. The initial testing of JMF on Intel and Solaris platforms look very promising.

Relationship to Other NIST Technical Activities

Currently, SSMM will be a standalone Java applet-based multimedia player running under any browser environment, which will has its own built-in synchronization and the ability to play audio and video streams. As SSMM becomes more mature, it could serve as a multimedia application to drive various network environments. As a result, it is very possible to have collaboration with Internetworking Group for IPv6 and High Speed Networking Group for ATM.

References

Ref.	Description	URL
1	Real-time Streaming Protocol	*ftp://ftp.ietf.org/internet-drafts/draft-ietf-mmusic-rtsp-06.txt*
2	Payload API Calls	*[TBD]*
3	RTP payload format for MPEG1/MPEG2 video (rfc2038)	*ftp://ds.internic.net/rfc/rfc2038.txt*
4	RTP payload format for H.261 video stream (rfc2032)	*ftp://ds.internic.net/rfc/rfc2032.txt*
5	MPEG-4 standard homepage	*http://drogo.cselt.stet.it/mpeg/standards/mpeg-4.htm*
6	MHEG-5 (Multimedia & Hypermedia Experts Group) introduction page	*http://www.fokus.gmd.de/ovma/mug/archives/documents/mheg-reader/rd1206.html*
7	RTP usage with Layered Multimedia Streams (draft)	*ftp://ftp.ietf.org/internet-drafts/draft-speer-avt-layered-video-02.txt*
8	A Transport Protocol for Real-Time Applications	*ftp://ds.internic.net/rfc/rfc1889.txt*
9	Resource ReSerVation Protocol	*ftp://ftp.ietf.org/internet-drafts/draft-ietf-rsvp-spec-16.txt*
10	ATM QoS Classes ??	*[TBD]*
11	middleware	*[TBD]*
12	SMIL (Synchronized Multimedia Integrated Language) of W3C	*http://www.w3.org/TR/PR-smil*